TNERec: Topic-Aware Network Embedding for Scientific Collaborator Recommendation

Conference Paper · October 2018

DOI: 10.1109/SmartWorld.2018.00177

6 authors, including:

Xiangjie Kong
Zhejiang University of Technology
138 PUBLICATIONS **2,231** CITATIONS
SEE PROFILE

Mengyi Mao
Dalian University of Technology
3 PUBLICATIONS **34** CITATIONS
SEE PROFILE

Bo Xu
Dalian University of Technology
83 PUBLICATIONS **436** CITATIONS
SEE PROFILE

Qun Jin
Waseda University
291 PUBLICATIONS **1,743** CITATIONS
SEE PROFILE

TNERec: Topic-aware Network Embedding for Scientific Collaborator Recommendation

Xiangjie Kong
School of Software,
Dalian University of Technology
Dalian, China

Mengyi Mao
School of Software,
Dalian University of Technology
Dalian, China

Jiaying Liu
School of Software,
Dalian University of Technology
Dalian, China

Bo Xu*
School of Software,
Dalian University of Technology
Dalian, China
Email: boxu@dlut.edu.cn

Runhe Huang
Hosei University
Japan

Qun Jin
Waseda University
Japan

***Abstract*—Collaboration is increasingly becoming a vital factor in an academic network, which can bring lots of benefits for scholars. Ubiquitous intelligence also provides an effective way for scholars to find collaborators. However, due to the large-scale of scholarly big data, there is a lot of information hard to capture in networks and we need to dig out valid information from collaboration networks. It is a valuable and urgent task to find appropriate collaborators for scholars. To address these problems, we hypothesize that fusing topic model and structure information could improve the performance of recommendations. In this paper, we propose a collaborator recommendation system, named TNERec (Topic-aware Network Embedding for scientific collaborator Recommendation), learning representations from scholars' research interests and network structure. TNERec first extracts scholars' research interests based on topic model and then learns vectors of scholars with network embedding. Finally, top-k recommendation list is generated based on the scholar vectors. Experimental results on a real-world dataset show the effectiveness of the proposed framework compared with state-of-the-art collaboration recommendation baselines.**

***Index Terms*—Topic Modeling, Network Embedding, Collaborator Recommendation**

I. Introduction

In today's academia, scholars prefer to collaborate with others rather than work alone. With the cost of communication decreasing and the development of ubiquitous intelligence, the forms of collaboration are more and more diverse. Collaboration can bring different perspectives to research subjects, and may result in unprecedented inspirations and breakthroughs. Scholars can exchange different ideas and share experience, expertise, and resources with each other. Previous studies show that collaborative scholars can bring out more productivities than those solo researchers [1], [2]. There are lots of variable factors contributing to the collaboration, e.g., academic level, research interests, and academic age [3]. Thus, it is important to find appropriate collaborators for each scholar in order to get more publications not only in quantity but also in quality.

With the rapidly growing of the massive academic data, large volume of scientific papers are produced. The millions of papers, authors, citations and other related data lead to the information overload and data deluge [4], [5] which are hard to compute. On the one hand, the situation brings the scholars with chances that they can acquire more information and resources conveniently, but on the other hand, it also induces the challenges. Among the massive scholars and complex collaboration networks, it is hard to find appropriate collaborators. Thus, the recommendation systems are useful to solve these problems in a certain extent and provide effective ways for scholars to find the relevant resources.

Existing collaborator recommendations can be divided into three categories including network-based, topic-based, and hybrid recommendations. Network-based recommendations calculate the similarity of nodes based on collaboration network to recommend collaborators for scholars, e.g., Common Neighbours, Jaccard Similarity, and global path index Katz. Another common network-based recommendation technique is Random Walk model. Although these works focus on the network structure extracted from collaboration network, each scholar still has a rich set of features need to be captured. Topic-based recommendations are to extract scholars' topics as research interests based on topic models for recommendation. Topic models extract a set of topics as the representation for documents, and these topics can be considered as research interests for scholars. However, it is still an challenge to find appropriate collaborators only through research interests. Thus, those hybrid recommendations are coined. Meanwhile, network embedding as an efficient method [6]–[19] aims to get low dimensional latent vectors from topological structure in graphs. Compared with learning from pure networks by network embedding, a combination of network topology and relevant information (e.g., research interests) needs to be leveraged. The studies on Attributed Network Embedding

This work was partially supported by the Fund for Promoting the Reform of Higher Education by Using Big Data Technology, Energizing Teachers and Students to Explore the Future (2017A01002), and the Fundamental Research Funds for the Central Universities (DUT18JC09).

(ANE) [7], [16] aim to leverage both network proximity and node attributes. Inspired by the idea of ANE, we combine the network structure and research interests and propose TNERec which is **T**opic-aware **N**etwork **E**mbedding framework for scientific collaborator **Rec**ommendation.

The main idea of TNERec is to fuse scholars' research topics with structure information in a unified representation for scholars. TNERec first obtains the research interests of scholars based on paper content and then learns the scholar vector based on collaboration network with attributes using network embedding methods. Finally, the similarities between scholars can be calculated by the cosine similarity between scholar vectors. Our experiments on a real-world dataset APS (American Physical Society) show the improvement of recommendation combining topic model with network embedding. The main contributions of this paper are summarized as follows:

- We present how to extract research interests for scholars based on paper content with topic model and build an attribute network with research interests.
- We propose TNERec, which learns representation of scholars combining research interests and network structure to generate better recommendation results.
- We conduct extensive experiments on a subset of APS dataset to evaluate the performance of our proposed method compared with RWR-based, topic-based, network-based methods. Promising results are presented and analyzed.

II. Related Work

A. Collaborator Recommendation

In the age of information, the academic collaboration is becoming more and more common. The data shows that the collaborative papers published in science, technology, and engineering are much more than those papers which are finished independently [20]. The collaborator recommendations can help scholars to find appropriate collaborators.

Random walk is the most common recommendation technique in the area of collaborator recommendation. The key of random walk is to dig out the similar structure of network based on probability. Xia et al. [21] proposed MVCWalker based on random walk with restart to recommend the most relevant collaborators for scholars. They considered three factors including coauthor order, latest collaboration time point and times of collaboration to calculate the importance of links between scholars. Zhou et al. [22] defined two measures which are sequence important measure and freshness importance measure to evaluate the importance of nodes and used random walk with restart in the heterogeneous network. However, random walk can only extract part of information from network.

Research interests are also exploited to recommend collaborators. In [23], Beneficial Collaborator Recommendation model was proposed considering topic distribution of research interest, interest variation with time, and researchers' impact in coauthor network. Tang et al. [24] proposed a cross-domain collaboration recommendation which considered topic layers and relevant topics to alleviate the sparseness issue and topic skewness for interdisciplinary collaborations.

Besides, some hybrid recommendation approaches have been proposed. Chaiwanarom and Lursinsap [25] proposed a hybrid algorithm based on dynamic collaboration over time which considered eight measures for interdisciplinary collaborator recommendation. Kong et al. [26] proposed a novel collaborator recommendation model combining the publication contents and collaboration network. They utilized a topic clustering model and a random walk model to identify the potential collaborators for researchers. Yang et al. [27] generated the research expertise, researchers' institutional connectivity and network proximity through SVM-rank fusion strategy. However, most of these recommendation techniques need manually-designed features, and it is not only inflexible but also time-consuming.

B. Topic Modeling

Topic models are widely applied for a large number of text documents in the field of natural language processing. All documents in corpus can be presented through several topics, then we can cluster these documents. According to [28], the methods of topic modeling can be divided into four categories: Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), and Correlated Topic Model (CTM).

Latent Dirichlet Allocation (LDA) is one of the most popular topic modeling approaches. LDA [29] is a three-level hierarchical Bayesian model for identifying latent topics according to the frequency of words from text corpora. The main idea of LDA is that each document can be represented as a mixture of several topics where each topic is a multinomial distribution defining the probability of words. There are a lot of works based on LDA. Yang et al. [30] proposed a weighted topic model for complementary collaborator recommendation. They first measured the research quality, and then adapt weighted probabilistic topic model based on LDA into a greedy algorithm to recommend collaborators. Sukhja et al. [31] analysed big data sets based on Spark LDA across multiple computing nodes for exploring sociological questions. Bian and Zhang [32] proposed a nonparametric hierarchical model based on triangle motif and topic model considering both networks data and node's text information to represent the node in network. Gopalan et al. [33] proposed collaborative topic Poisson factorization (CTPF) which is also based on LDA and is considered in section III-A. Wang and Blei [34] developed an algorithm to recommend papers for users. They combined the merits of traditional collaborative filtering and probabilistic topic modeling based on LDA. However, this kind of recommendation technique ignores the network information and needs more computing time and data processing. In our work, we apply topic model to extract scholars' research interests.

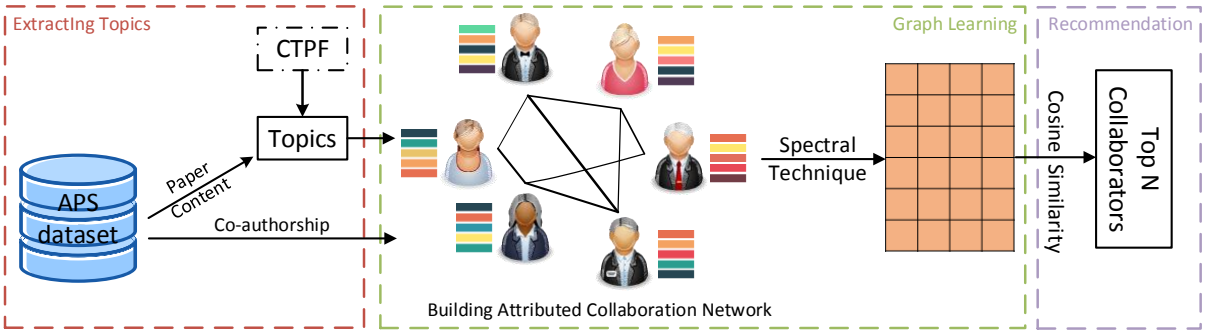

Fig. 1. The structure of TNERec: depicts the three main components including topic extracting, graph learning, and recommendation.

C. Network Embedding

Network embedding is originated from embedding learning techniques, e.g., Laplacian Eigenmaps [35], LLE [36], IsoMAP [37]. In recent years, network embedding gains a lot of attentions and popularity. Network embedding aims to represent a network with millions of nodes and edges as continuous low-dimensional vectors and preserve network structure [14], [15]. After representing the network as vectors, noise or redundant information can be reduced and more methods can be directly applied to large-scale network computation to solve the graph analytics problem. Thus, Network embedding is capable of supporting a variety of works such as node classification, node clustering, network visualization, network compression, and link prediction [13]. Up till now, a lot of network embedding algorithms and models have been proposed. For example, DeepWalk [6] learns latent representations of nodes using skip-gram model based on the node sequences extracted from random walk. Based on DeepWalk, node2vec [10] extends the biased random walks and preserves higher-order proximity which can be applied to the weighted network.

However, the networks in real world are more complicated, it's far from enough to transform structural information into vertex embeddings. Some works focus on integration of text information into conventional network embedding methods. Yang et al. [9] proposed text-associated DeepWalk (TADW) which combines the text information and network structure under the framework of matrix factorization. Sun et al. [11] proposed content-enhanced network embedding (CENE). They considered the text content as node-content link, and then optimized the joint objective function of both text information and network structure. Ganguly and Pudi proposed Paper2vec [18] which combines the doc2vec with DeepWalk to capture the text information of papers and citation network structure. Besides, attributed network embedding is also a hot spot of research. Chang et al. [7] considered three modules (e.g., image-image, image-text, and text-text) and exploited a highly nonlinear multi-layered embedding function to capture different kinds of data in the heterogeneous network. LANE [16] combined the labels with the attributed network embedding. However, few works applied network embedding to collaborator recommendations.

Algorithm 1 TNERec

Input: paper content PD, rating Q, collaboration network G, topics number TN, attributes number AN, iteration times IT, representation dimension D
Output: recommendation list RL

TD = CTPF (PD, Q, TN, IT);
for $i = 1$, $i++$ **do**
  A_i = Top (TD_i, AN);
end for
Construct affinity matrices C_G, C_A;
Compute Laplacian matrices L_G, L_A;
$t = 0$, $R_G = 0$, $R_A = 0$, $S = 0$;
while $t < IT$ **do**
  Update R_G, R_A, S according to Equation 9;
  $t = t + 1$;
end while
for $i = 1$, $i++$ **do**
  $RL[i]$ = most k similar scholars;
end for

III. Methods

In this section, we introduce the framework of TNERec (in Figure 1). For a given collaboration network $G = (V, E, W)$, V is the set of nodes, each edge $e_{i,j} \in E$ denotes collaboration relationship between two scholars v_i and v_j, and $w_{i,j} \in W$ represents the weight of relationship (we consider the collaboration times in this paper) between scholar v_i and v_j. We extract the research topic information of each scholar v_i through topic model. These topics make up scholar's attributes a_i. We combine the collaboration network G with nodes' attributes A into $G' = \{G, A\}$. Thus, we consider G' as a undirected but weighted collaboration network with nodes' attributes. What we want is to get a lower dimensional representation S for each scholar based on the graph G', where vector $s_i \in \mathbb{R}^{|D|}$ and $|D| \ll |V|$. Matrix S is supposed to occupy a low-dimensional latent space and maintain the original topological information of network as well as scholars' research topics.

Based on $s_i \in S$, we calculate the similarity and choose the top-k similar scholars for recommendation. The algorithm of TNERec is described in Algorithm 1.

A. Extracting Research Topics

First, we want to get each scholar's topics from text information of publications and user ratings. The text information of scholars can be considered as the papers they published including the papers titles, abstracts or the main body of the full papers. The user ratings are the relationship between scholars and papers, which can be regarded as an adjacency matrix. The rating r_{v_i,p_j} equals one if scholar v_i published paper p_j, and is zero otherwise. Based on CTPF model [33], for each scholar v_i we can get his/her associated topic distribution TD_{v_i}.

In order to get a topic distribution for each scholar v_i, Equation 1 shows how to fit a CTPF posterior distribution over variables:

$$P(\eta_{v_i}|\boldsymbol{w}, \boldsymbol{r}), \tag{1}$$

where $\boldsymbol{w}$ is the set of the context of papers (in the form of bags of words) and $\boldsymbol{r}$ is user ratings for papers. η_{v_i} is a vector presenting the preference of scholar v_i for each topic, and the latent of η is the number of topics we can defined. Equation 1 can be resolved by the posterior probability over four variables,

$$P(\theta, \beta, \eta, \epsilon|\boldsymbol{w}, \boldsymbol{r}). \tag{2}$$

θ_{p_j} denotes the distribution of topics for paper p_j which can be calculated by LDA. ϵ_{p_j} is topic offsets of paper p_j that capture the deviation of paper p_j from topics in θ_{p_j}. β_t is the distribution of words for topic t which can also be calculated by LDA. Then, the main job is to optimize these parameters to find a distribution to minimize Kullback-Leibler divergence of Equation 2. We introduce two new variables ϕ and ξ to solve the problem. ϕ is added to present the relationship among papers, words, and topics. ξ is added to explore the relationship among scholars, papers, and topics.

$$\begin{cases} \phi_{p_j,w} = \log \theta_{p_j,t} + \log \beta_{w,t}, \\ \xi_{v_i,p_j} = \begin{cases} \log \eta_{v_i,t} + \log \theta_{p_j,t}, & t < T, \\ \log \eta_{v_i,t} + \log \epsilon_{p_j,t}, & T \le t < 2T, \end{cases} \end{cases} \tag{3}$$

where T is the number of topics. $\phi_{p_j,w}$ is a T-dimensional Poisson counts, while ξ_{v_i,p_j} is a $2T$-dimensional Poisson counts including $\xi^a_{v_i,p_j}$ and $\xi^b_{v_i,p_j}$. The complete iteration of $\theta, \beta, \eta, \epsilon$ are shown:

$$\begin{cases} \theta_{p_j,t} = a + \sum_w \phi_{p_j,w,t} + b \cdot \sum_{v_i} \xi^a_{v_i,p_j,t} + \sum_w \beta_{w,t} + \sum_{v_i} \eta_{v_i,t}, \\ \beta_{w,t} = c + d \cdot \sum_{p_j} \phi_{p_j,w,t} + \sum_{p_j} \theta_{p_j,t}, \\ \eta_{v_i,t} = e + \sum_{p_j} \xi^a_{v_i,p_j,t} + f \cdot \sum_{p_j} \xi^b_{v_i,p_j,t} + \sum_{p_j} (\theta_{p_j,t} + \epsilon_{p_j,t}), \\ \epsilon_{p_j,t} = g + h \cdot \sum_{v_i} \xi^b_{v_i,p_j,t} + \sum_{v_i} \eta_{v_i,t}, \end{cases} \tag{4}$$

These four parameters (in Equation 4) are Gamma distributions. $\theta_{p_j,t}$ follows $Gamma(a, b)$, $\beta_{w,t}$ follows $Gamma(c, d)$, $\eta_{v_i,t}$ follows $Gamma(e, f)$, and $\epsilon_{p_j,t}$ follows $Gamma(g, h)$. Equation 3 and Equation 4 can be optimized with a coordinate ascent algorithm, and a parameter in Equation 4 is updated while other parameters remain fixed. For each scholar v_i, we can get the topic distribution TD_{v_i} from Equation 1 based on Equation 3 and Equation 4. Each topic from distribution TD_{v_i} can be considered as an attribute for each scholar. For example, we define 30 topics for all documents, and each scholar has a distribution about these 30 topics. Then, we select the most relevant N topics from these 30 topics and assign different weights for N topics according to the relativity (i.e., we assign 5 for the most relevant topic and assign 1 for the less relevant topic if we select 5 topics for each scholar). These N topics can be weighted-attributes $a_i = \{(t_1, w_{t_1}), (t_2, w_{t_2}), \cdots, (t_n, w_{t_n})\}$ for scholar v_i. Figure 2 illustrates the process of extracting scholars' topic attributes.

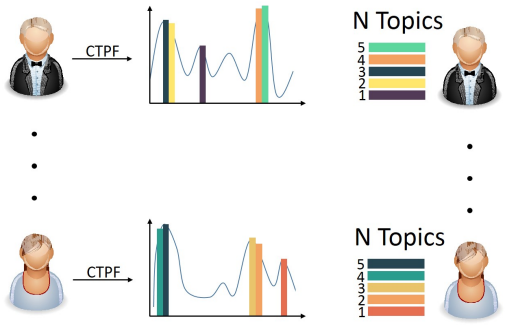

Fig. 2. A process of extracting scholars' topic attributes. We first get topic distribution from CTPF and then select the most relevant N topics as scholars' weighted-attributes.

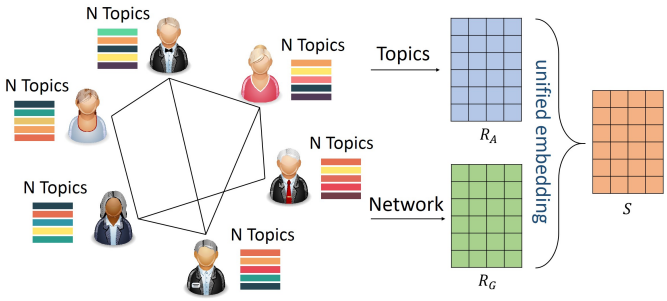

Fig. 3. An illustration of the network embedding for attributes. First, apply spectral embedding on the collaboration network and topic attributes, and obtain two embeddings R_G and R_A. Then, optimize a unified embedding representation S for scholars according to R_G and R_A correlation.

B. Network Embedding for Attributes

After gaining the research topics of each scholar and the topological structure of collaboration network, we want to integrate two kinds of information to represent a scholar. Inspired by LANE [16], we first need to get the representation R_G for network structure and R_A for topic attributes, respectively. Then a unified embedding representation S for scholars we want is learned jointly from two representations R_G and R_A with both correlations. Figure 3 shows the main idea of network embedding for attributes. First, the main work is to seek R_G and R_A. The aim of representation R_G is to preserve more structure information of network G with lower

dimension. In other words, if v_i and v_j have similar topological structure, their representations r_i and r_j should be similar. A rational choice of R_G according to [35] is to meet the constraints in Equation 5,

$$\min_{R_G} \quad \frac{1}{2}\sum_{i,j=1}^{n} c_{i,j}||\frac{r_i}{\sqrt{d_i}} - \frac{r_j}{\sqrt{d_j}}||_2^2, \tag{5}$$

where $c_{i,j} = g_i^{\mathrm{T}} g_j$ ($c_{i,j} \in C_G$) is the similarity of nodes i and j, and G is the adjacency matrix for network. D_G ($D_G(i,i) = d_i = \sum_{j=1}^{n} c_{i,j}$) is diagonal matrix where each element is the sum of corresponding row of C_G. Equation 5 can be converted to objective function 6 based on normalized graph Laplacian.

$$\begin{aligned} \max_{R_G} \quad & f_G = \mathrm{Tr}(R_G^{\mathrm{T}} L_G R_G), \\ \mathrm{S.T.} \quad & R_G^{\mathrm{T}} R_G = \mathrm{I}, \end{aligned} \tag{6}$$

where laplacian $L_G = D_G^{-\frac{1}{2}} C_G D_G^{-\frac{1}{2}}$. The way to get the attribute representation R_A is similar to R_G with objective function f_A.

$$\begin{aligned} \max_{R_A} \quad & f_A = \mathrm{Tr}(R_A^{\mathrm{T}} L_A R_A), \\ \mathrm{S.T.} \quad & R_A^{\mathrm{T}} R_A = \mathrm{I}, \end{aligned} \tag{7}$$

where laplacian $L_A = D_A^{-\frac{1}{2}} C_A D_A^{-\frac{1}{2}}$, $c'_{i,j} = a_i^{\mathrm{T}} a_j$ ($c'_{i,j} \in C_A$), $D_A(i,i) = d'_i = \sum_{j=1}^{n} c'_{i,j}$, and A is the adjacency matrix for topic attributes. According to spectral graph theory [35], [38], [39], spectral techniques can identify arbitrary shape graphs and converge to a global optimal solution, besides, spectral techniques are insensitive to noise relatively in the graph.

To capture G and A their interdependency and to complement each other, our goal is to combine them and maximize their correlations to obtain the unified representation S through two parameters to balance the weight of topic attributes and the weight of correlations between G and A according to [16].

$$\begin{aligned} \max_{R_G, R_A, H} \quad & f = [f_G + \alpha_1 f_A + \alpha_2 \mathrm{Tr}(R_A^{\mathrm{T}} R_G R_G^{\mathrm{T}} R_A)] \\ & + \mathrm{Tr}(R_G^{\mathrm{T}} S S^{\mathrm{T}} R_G) + \mathrm{Tr}(R_A^{\mathrm{T}} S S^{\mathrm{T}} R_A), \\ \mathrm{S.T.} \quad & R_G^{\mathrm{T}} R_G = \mathrm{I}, R_A^{\mathrm{T}} R_A = \mathrm{I}, S^{\mathrm{T}} S = \mathrm{I}. \end{aligned} \tag{8}$$

α_1 is the weight for topic attributes and α_2 is the weight for the correlations between R_G and R_A. $\mathrm{Tr}(R_A^{\mathrm{T}} R_G R_G^{\mathrm{T}} R_A)$ is added to measure the correlation between R_G and R_A, and make them to compensate each other. The matrices $\mathrm{Tr}(R_G^{\mathrm{T}} S S^{\mathrm{T}} R_G)$ and $\mathrm{Tr}(R_A^{\mathrm{T}} S S^{\mathrm{T}} R_A)$ are added to estimate the correlations among S, R_G, and R_A. Thus, S can preserve the network and topic attributes information to a greater extent.

By solving Equation 8, we can get representation S with more structure and attributes information. However, there are many variables in Equation 8, first we can reformulate Equation 8 with Lagrangian function via Lagrange multipliers, then calculate the second derivative of each variable with others fixed. We are able to get the local maximum for each variable. Equation 8 can be converted to Equation 9:

$$\begin{cases} (L_G + \alpha_2 R_A R_A^{\mathrm{T}} + S S^{\mathrm{T}}) R_G = \lambda_1 R_G, \\ (\alpha_1 L_A + \alpha_2 R_G R_G^{\mathrm{T}} + S S^{\mathrm{T}}) R_G = \lambda_2 R_A, \\ (R_G R_G^{\mathrm{T}} + R_A R_A^{\mathrm{T}}) S = \lambda_3 S, \end{cases} \tag{9}$$

where $\lambda_i (i = 1, 2, 3)$ denotes the Lagrange multipliers for R_G, R_A, or S. In this case, the problem comes down to the generalized eigen-problem $L_{G/A/S}\mathbf{y} = \gamma D_{G/A/S}\mathbf{y}$. Let the solution is $\{y_0, y_1, \cdots, y_n\}$, which is ordered by their eigenvalues $\gamma_0 \leq \gamma_1 \leq \cdots \leq \gamma_n$. In fact, γ_0 is zero, and as we known that the only eigenvector of γ_0 is $\mathbf{1}$. However, we do not need the eigenvector $\mathbf{1}$, so we take top D eigenvectors from y_1, i.e., $R_G = [y_1, y_2, \cdots, y_D]$. Through multiple iterations, we can get representation S for scholars with the latent space $\mathbb{R}^{|D|}$.

C. Collaborator Recommendation

We want to recommend the most similar collaborators for each scholar. After obtaining the vector representation for each scholar, we can calculate the cosine similarity between each pair of scholars based on the vector s,

$$sim(v_i, v_j) = \frac{s_i \cdot s_j}{\sqrt{|s_i| \cdot |s_j|}}. \tag{10}$$

According to Equation 10, we can acquire most k-similar collaborators. Top-k recommendation list is generated for each scholar.

IV. Experiments

In this section, we conduct experiments to evaluate the performance of our proposed framework TNERec for recommendation on real-world dataset. We introduce the dataset and experimental settings before presenting the results of our experiments.

A. Dataset

We utilized PRB (Physical Review B) dataset from APS[1] (American Physical Society). For APS dataset, we first had name disambiguation on authors from 1893 to 2015 based on [40]. We removed authors who have less than 2 collaborators from 2006 to 2010. We built a collaboration network based on the rest of authors and relationships and found the maximal connected component subgraphs of the network. Finally, We extracted 34,905 scholars and 14,055 papers including 331,014 coauthor relationships. Each scholar is regarded as a node, every edge between two scholars indicates a coauthor relationship. We divided dataset into a training set and a test set to validate the effectiveness of TNERec. We randomly selected coauthor relationships with the ratio R into the training set and preserved as many scholars as possible in the training set. The text information of each scholar's topics that is needed in Section III-A includes paper titles of every paper.

B. Baseline Methods

We employ the following methods as baseline methods.

1) MVCWalker: MVCWalker [21] is a random walk model for collaborator recommendation which considers three academic factors including coauthor order, latest collaboration time, and times of collaboration as link importance.

[1]https://journals.aps.org/datasets

2) CTPF: CTPF [33] is probabilistic model of articles to represent scholars with their preferences for topics. It integrates two ideas: collaborative topic regression and Poisson factorization.

3) TNERec-G: TNERec-G is a variation of the proposed TNERec with only collaboration network information.

In MVCWalker, we set the iteration times to 25 and damping coefficient to 0.3 which determines the probability of walking to the next neighbour. In CTPF, we set the number of topics to 30 and iteration times to 10, and we recommend scholars based on the extracted topics for each scholar from CTPF. In TNERec-G and TNERec, we set the iteration times to 10 and the dimension of the embedding representation to 100. We choose the most relevant 10 topics from 30 topics extracted by CTPF as the attributes for each scholar in TNERec.

C. Evaluation Metrics

We employed four evaluation metrics to investigate the effectiveness of TNERec: Precision, Recall, F1 and NDCG (Normalized Discounted Cumulative Gain).

$$Precision = \frac{1}{m}\sum_{i=1}^{m}\frac{K_i}{RL} \tag{11}$$

$$Recall = \frac{1}{m}\sum_{i=1}^{m}\frac{K_i}{TS} \tag{12}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{13}$$

$$NDCG = \frac{1}{m}\sum_{i=1}^{m}(Z_k \sum_{j=1}^{k}\frac{2^{r(j)}-1}{log(1+j)})_i \tag{14}$$

Precision is the ratio of the recommended collaborators who are in the test set to all candidates corresponding to their target scholars. Recall is the ratio of the recommended collaborators who are in the test set to all collaborators. F1 is a comprehensive evaluation index and is harmonic mean of Precision and Recall. NDCG is a metric to measure the ranking quality for giving higher score to the top items. K_i in Equation 11 and 12 denotes the number of the recommended collaborators for scholar v_i who are collaborators of scholar v_i in the test set. RL in Equation 11 is the length of recommendation list and TS in Equation 12 is the length of test set. In Equation 13, Z_k is normalization to make $(Z_k \sum_{j=1}^{k}\frac{2^{r(j)}-1}{log(1+j)})_i$ range from 0 to 1 and $r(j)$ is to assess relevance. We define $r(j) = 1$ which represents the recommended collaborator is in the test set, and $r(j) = 0$ otherwise.

D. Results Analysis

1) Parameter Study: We conduct the parameter analysis of TNERec on two parameters: the weight for topic attributes α_1 and the weight for the correlations between R_G and R_A α_2. We set the vector dimension fixed as 100, the ratio of training set R as 30%, the length of recommendation list k as 5 and choose five pairs parameters of α_1 and α_2 to find some laws of the balance between topic attributes and network structure.

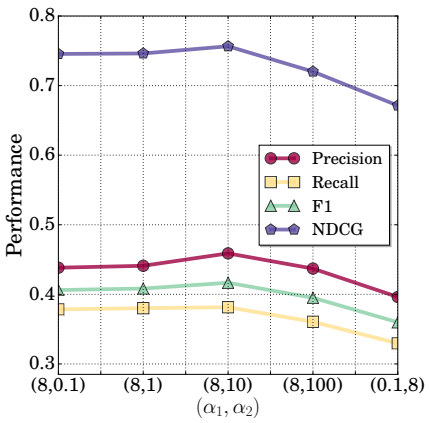

Fig. 4. Performance of TNERec with different parameters α_1 and α_2. We set the dimension of the embedding representation as 100, the length recommendation list k as 5, and the ratio of training set R as 30%.

Figure 4 shows the comparison results of TNERec where (α_1, α_2) equals to $(8, 0.1)$, $(8, 1)$, $(8, 10)$, $(8, 100)$, $(0.1, 8)$, respectively. Among the first four set of (α_1, α_2), α_1 is fixed as 8 and α_2 range from 0.1 to 100 with the 10-fold increase. From figure 4, we can find out that the performance of TNERec is best when $(\alpha_1, \alpha_2) = (8, 10)$. The less difference between α_1 and α_2 is, the better the performance of TNERec is. What's more, the value of α_1 can't be too small. For example, there is a little difference between $(8, 0.1)$ and $(0.1, 8)$ except for the values of α_1 and α_2 exchanged, but the performance of TNERec differs a lot. Because the value of α_2 is a bit large and much larger than α_1, the result of $(\alpha_1, \alpha_2) = (8, 100)$ is also unsatisfactory. Thus, topic attributes have an effect on improvement of TNERec and network structure also needs to be preserved.

2) Influence of Recommendation list: We carried out experiments to evaluate the performance of TNERec with the increase of the length of recommendation list k. We choose the ratio of training set 30% to generate the recommendation list and set the dimension of the embedding representation as 100 for TNERec and TNERec-G. The parameters α_1 and α_2 are set as 8 and 10, respectively. Figure 5 shows the comparisons with other methods in terms of precision, recall, F1, and NDCG. With the increase of recommendation list k, we can find out that the precision of TNERec, CTPF, and TNERec-G decreases while the precision of MVCWalker tends to increase at first then decrease. The recall of all methods shows the upwards trend. F1 of all methods takes on a tendency of increasing first but decreasing afterwards with the recommendation list increasing, but the highest point of the curves are different. The NDCG of TNERec, CTPF, and TNERec-G stays stable when $k \geq 5$, while NDCG of MVCWalker increases a lot when $k \leq 3$. What's more, TNERec outperforms the others in NDCG by a large margin. When the number of recommended collaborators is not large, TNERec and CTPF are more advantageous. TNERec has the advantage of the high quality recommendation because the

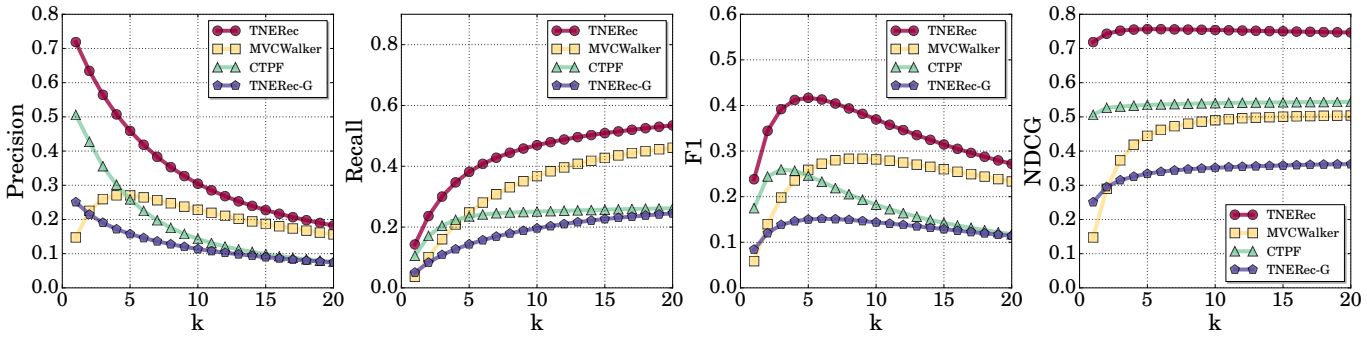

Fig. 5. Comparison between TNERec and other methods in recommendation quality over different length of recommendation list k. We set the ratio of training set R as 30% for all methods, the dimension of the embedding representation as 100 for TNERec and TNERec-G, and α_1 as 8 and α_2 as 10 for TNERec.

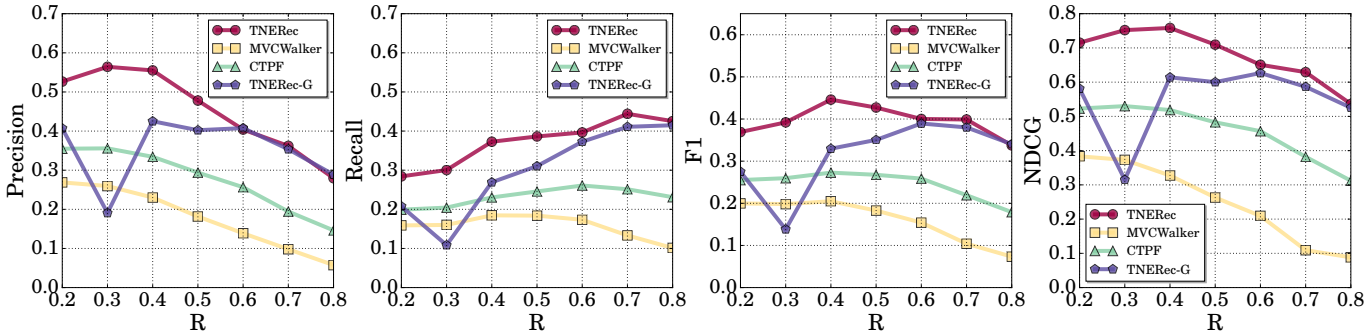

Fig. 6. Comparison between TNERec and other methods in recommendation quality over different ratio of training set R. We set the length of recommendation list k as 3 for all methods, the dimension of the embedding representation as 100 for TNERec and TNERec-G, and α_1 as 8 and α_2 as 10 for TNERec.

correct recommended collaborators in the top of recommendation list. TNERec outperforms other baseline methods over all length of recommendation list on these four metrics.

3) Influence of Training set: We change the size of training set to evaluate the performance of TNERec over the training set. The ratio of training set R ranges from 20% to 80%. We set the length of recommendation list k as 3 and the dimension of the embedding representation as 100 for TNERec and TNERec-G. The parameters α_1 and α_2 are set as 8 and 10, respectively. Figure 6 depicts the comparisons with other methods over different training set in terms of precision, recall, F1, and NDCG. From figure 6, we can find out that the precision and NDCG of TNERec, MVCWalker, and CTPF show downtrends in general with the increase of the ratio of training set R. The recall of TNERec increases while the recall of MVCWalker and CTPF stays stable with the size of training set increasing. F1 of TNERec, MVCWalker, and CTPF takes on a tendency of increasing first but decreasing afterwards. Surprisingly, the performance of TNERec-G drops a lot on the ratio of training set 30%, but shares the similar trends with TNERec on other size of training set. TNERec outperforms all other methods a lot on four metrics when the ratio of training set is smaller than 50%. TNERec seems to have a better performance on the smaller training set, and the added topic attributes show their advantages when dealing with the incomplete collaboration network.

V. Conclusions

Facing with the challenge of information overload, it is difficult to select appropriate collaborators from a huge number of scholars. In this paper, we focus on recommending collaborators for scholars based on network representation and topic model when scholars want to search for collaborators on the scientific social platforms. To this end, we propose a novel framework, TNERec, which considers both scholars' research interests and collaboration network structure. TNERec first extracts topics from paper content to find the scholars' latent research interests and then treats these topics as attributes of nodes in the collaboration network to learn embeddings for scholars. According to the scholar vectors based on network embeddings, TNERec calculates the cosine similarity for recommendation. Finally, Experiments on a real-world APS dataset demonstrate that TNERec consistently performs the most effective compared with baseline methods in terms of precision, recall, F1, and NDCG.

There is still room for future studies in this direction. We only count on paper content and network structure while many other entities like journals and affiliations in the heterogeneous networks can be studied for collaborator recommendation.

Thus, we plan to extend TNERec for more attributes to enhance the embeddings and test the effectiveness of TNERec on more datasets.

REFERENCES

[1] S. Lee and B. Bozeman, "The impact of research collaboration on scientific productivity," *Social Studies of Science*, vol. 35, no. 5, pp. 673–702, 2005.

[2] S. Wuchty, B. F. Jones, and B. Uzzi, "The increasing dominance of teams in production of knowledge," *Science*, vol. 316, no. 5827, pp. 1036–1039, 2007.

[3] W. Wang, S. Yu, T. M. Bekele, X. Kong, and F. Xia, "Scientific collaboration patterns vary with scholars' academic ages," *Scientometrics*, vol. 112, no. 1, pp. 329–343, Jul 2017.

[4] F. Xia, W. Wang, T. M. Bekele, and H. Liu, "Big scholarly data: A survey," *IEEE Transactions on Big Data*, vol. 3, no. 1, pp. 18–35, 2017.

[5] S. Khan, X. Liu, K. A. Shakil, and M. Alam, "A survey on scholarly data: From big data perspective," *Information Processing & Management*, vol. 53, no. 4, pp. 923–944, 2017.

[6] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: online learning of social representations," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701–710.

[7] S. Chang, W. Han, J. Tang, G.-J. Qi, C. C. Aggarwal, and T. S. Huang, "Heterogeneous network embedding via deep architectures," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '15. ACM, 2015, pp. 119–128.

[8] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*, ser. WWW '15. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[9] C. Yang, D. Zhao, D. Zhao, E. Y. Chang, and E. Y. Chang, "Network representation learning with rich text information," in *International Conference on Artificial Intelligence*, 2015, pp. 2111–2117.

[10] A. Grover and J. Leskovec, "Node2vec: Scalable feature learning for networks," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. ACM, 2016, pp. 855–864.

[11] X. Sun, J. Guo, X. Ding, and T. Liu, "A general framework for content-enhanced network representation learning," *arXiv preprint arXiv:1610.02906*, 2016.

[12] A. García-Durán and M. Niepert, "Learning graph representations with embedding propagation," *arXiv preprint arXiv:1710.03059*, 2017.

[13] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *arXiv preprint arXiv:1705.02801*, 2017.

[14] H. Cai, V. W. Zheng, and K. Chang, "A comprehensive survey of graph embedding: Problems, techniques and applications," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2018.

[15] P. Cui, X. Wang, J. Pei, and W. Zhu, "A survey on network embedding," *arXiv preprint arXiv:1711.08752*, 2017.

[16] X. Huang, J. Li, and X. Hu, "Label informed attributed network embedding," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, ser. WSDM '17, 2017, pp. 731–739.

[17] C. Tu, H. Liu, Z. Liu, and M. Sun, "CANE: Context-aware network embedding for relation modeling," in *Meeting of the Association for Computational Linguistics*, 2017, pp. 1722–1731.

[18] S. Ganguly and V. Pudi, "Paper2vec: Combining graph and text information for scientific paper representation," in *Advances in Information Retrieval*, J. M. Jose, C. Hauff, I. S. Altıngovde, D. Song, D. Albakour, S. Watt, and J. Tait, Eds. Cham: Springer International Publishing, 2017, pp. 383–395.

[19] X. Kong, M. Mao, W. Wang, J. Liu, and B. Xu, "VOPRec: Vector representation learning of papers with text information and structural identity for recommendation," *IEEE Transactions on Emerging Topics in Computing*, 2018, DOI:10.1109/TETC.2018.2830698.

[20] B. Bozeman and C. Boardman, "Research collaboration and team science: A state-of-the-art review and agenda," *Springerbriefs in Entrepreneurship & Innovation*, 2014.

[21] F. Xia, Z. Chen, W. Wang, J. Li, and L. T. Yang, "MVCWalker: Random walk-based most valuable collaborators recommendation exploiting academic factors," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 3, pp. 364–375, 2014.

[22] X. Zhou, L. Ding, Z. Li, and R. Wan, "Collaborator recommendation in heterogeneous bibliographic networks using random walks," *Information Retrieval Journal*, vol. 20, no. 4, pp. 317–337, Aug 2017.

[23] X. Kong, H. Jiang, W. Wang, T. M. Bekele, Z. Xu, and M. Wang, "Exploring dynamic research interest and academic influence for scientific collaborator recommendation," *Scientometrics*, vol. 113, no. 1, pp. 369–385, Oct 2017.

[24] J. Tang, S. Wu, J. Sun, and H. Su, "Cross-domain collaboration recommendation," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '12. ACM, 2012, pp. 1285–1293.

[25] P. Chaiwanarom and C. Lursinsap, "Collaborator recommendation in interdisciplinary computer science using degrees of collaborative forces, temporal evolution of research interest, and comparative seniority status," *Knowledge-Based Systems*, vol. 75, pp. 161 – 172, 2015.

[26] X. Kong, H. Jiang, Z. Yang, Z. Xu, F. Xia, and A. Tolba, "Exploiting publication contents and collaboration networks for collaborator recommendation," *PLOS ONE*, vol. 11, no. 2, pp. 1–13, 02 2016.

[27] C. Yang, J. Sun, J. Ma, S. Zhang, G. Wang, and Z. Hua, "Scientific collaborator recommendation in heterogeneous bibliographic networks," in *2015 48th Hawaii International Conference on System Sciences*, Jan 2015, pp. 552–561.

[28] R. Alghamdi and K. Alfalqi, "A survey of topic modeling in text mining," *International Journal of Advanced Computer Science & Applications*, vol. 6, no. 1, 2015.

[29] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J Machine Learning Research Archive*, vol. 3, pp. 993–1022, 2003.

[30] C. Yang, J. Ma, X. Liu, J. Sun, T. Silva, and Z. Hua, "A weighted topic model enhanced approach for complementary collaborator recommendation," in *Pacific Asia Conference on Information Systems, PACIS 2014*, 01 2014.

[31] N. Sukhija, M. Tatineni, N. Brown, M. V. Moer, P. Rodriguez, and S. Callicott, "Topic modeling and visualization for big data in social sciences," in *2016 Intl IEEE Conferences on Ubiquitous Intelligence Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, July 2016, pp. 1198–1205.

[32] X. Bian and K. Zhang, "Modeling network with topic model and triangle motif," in *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, Aug 2015, pp. 880–886.

[33] P. Gopalan, L. Charlin, and D. M. Blei, "Content-based recommendations with poisson factorization," in *International Conference on Neural Information Processing Systems*, 2014, pp. 3176–3184.

[34] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '11. ACM, 2011, pp. 448–456.

[35] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," *Advances in Neural Information Processing Systems*, vol. 14, no. 6, pp. 585–591, 2001.

[36] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[37] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[38] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: analysis and an algorithm," in *International Conference on Neural Information Processing Systems: Natural and Synthetic*, 2001, pp. 849–856.

[39] U. Luxburg, *A tutorial on spectral clustering*. Kluwer Academic Publishers, 2007.

[40] R. Sinatra, D. Wang, P. Deville, C. Song, and A.-L. Barabási, "Quantifying the evolution of individual scientific impact," *Science*, vol. 354, no. 6312, 2016.